UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/611,920	07/07/2000	David J. Lindner	2007.0012900	7765

22879 7590 02/21/2006

HEWLETT PACKARD COMPANY
P O BOX 272400, 3404 E. HARMONY ROAD
INTELLECTUAL PROPERTY ADMINISTRATION
FORT COLLINS, CO 80527-2400

EXAMINER	
SINGH, RACHNA	
ART UNIT	**PAPER NUMBER**
2176	

DATE MAILED: 02/21/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 09/611,920
Filing Date: July 07, 2000
Appellant(s): LINDNER, DAVID J.

Mark E. Scott
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 06/16/05 appealing from the Office action mailed 03/07/05.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

No amendment after final has been filed.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

5,864,852	LUOTONEN	01-1999

Cluet, Sophie, Olga Kapitskaia, and Divesh Srivastava, "Using LDAP Directory Caches", ACM Database, May 1999.

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claim Rejections - 35 USC § 101

35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claims 16-20, 33-38, and 42-44 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. The claims are directed to a "caching daemon". A daemon is a process that runs in the background and performs a specified operation at predefined times or in response to certain events and thus is directed to non-statutory subject matter since it does not define any structural and functional interrelationships between the computer program and other claimed elements of a computer which permit the computer program's functionality to be realized. A caching daemon is software per se.

Claim Rejections - 35 USC § 103

. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 1-20 are rejected under 35 U.S.C. 103(a) as being unpatentable over Cluet, Sophie, Olga Kapitskaia, and Divesh Srivastava, "Using LDAP Directory Caches", ACM database, May1999 in view of Luotonen, US 5,864,852, 1/26/99.

In reference to claim 1, Cluet teaches the use of LDAP directory caches. Cluet's system teaches that in order to achieve fast performance and high availability in LDAP network directories, it is desirable to cache information close to the applications that access the directory information. See page 273. Compare to ***"simultaneously maintaining a first plurality of connections between the directory server and a caching daemon"***. While Cluet does not delve into the details of the LDAP directory cache; Luotonen provides some insight. Luotonen teaches a system in which when a client requests a document, a proxy server determines if the file contained in the cache is up-to-date and delivers the document to the user if it is. If it is not up-to-date, the proxy server then it retrieves information from the origin server (directory server). See column 1, lines 39-50. Compare to ***"determining if an application is requesting information from the directory server; determining if the requested information is stored in the caching daemon in response to determining that the application has requested information; and sending the requested information to the application."*** Luotonen further cites another caching proxy server, the Harvest Cache Daemon, see column 2, lines 9-15. It would have been obvious to one of ordinary skill in the art at the time of the invention to combine Cluet and Luotonen since both are concerned with providing a cache daemon with a directory server to improve performance as providing caching helps achieve fast performance and high availability. See page 273 or Cluet and column 1, lines 25-35 of Luotonen where he recites "***the tremendous continuing growth of the Web makes it necessary to have intermediate servers which perform caching (store documents locally, such that***

documents may be quickly accessed from the local file system, instead of being retransferred again from the original server". Furthermore, providing a cache with a directory server not only improves performance but also avoids latency which is a major limitation when the cache is not connected to the directory server (as taught by both Cluet and Luotonen).

In reference to claim 2, Cluet teaches that the directory server is linked to a variety of network-based applications that store data and providing a cache storing those entries provides for efficient delivery of documents. Thus the directory server is not limited in its connections between various applications and a caching daemon. See page 273.

In reference to claims 3-5, Luotonen teaches that when a client requests a document, a proxy server determines if the file contained in the cache is up-to-date and delivers the document to the user if it is. If it is not up-to-date, the proxy server then it retrieves information from the origin server (directory server). See column 1, lines 39-50. Thus if the document were not stored in the cache originally, the proxy would then retrieve it from the remote server (directory server). The information is then stored in the proxy for future requests. See column 1. It would have been obvious to one of ordinary skill in the art at the time of the invention to combine Cluet and Luotonen since both are concerned with providing a cache daemon with a directory server to improve performance as providing caching helps achieve fast performance and high availability. See page 273 or Cluet and column 1, lines 25-35 of Luotonen.

Claims 11-15 are rejected under the same rationale used in reference to claims 1-5 respectively above.

In reference to claims 6 and 8, Cluet teaches the use of LDAP directory caches. Cluet's system teaches that in order to achieve fast performance and high availability in LDAP network directories, it is desirable to cache information. See page 273. While Cluet does not delve into the details of the LDAP directory cache; Luotonen provides some insight. Luotonen teaches a system in which when a client requests a document, a proxy server determines if the file contained in the cache is up-to-date and delivers the document to the user if it is. If it is not up-to-date, the proxy server then it retrieves information from the origin server (directory server). See column 1, lines 39-50. Compare to ***"a directory server for storing information; and a caching daemon adapted to establish a first plurality of connections to the directory server, determine if an application is requesting information from the directory server, determine if the requested information is stored within the caching daemon; and send the requested information to the application."*** Luotonen further cites another caching proxy server, the Harvest Cache Daemon, see column 2, lines 9-15. It would have been obvious to one of ordinary skill in the art at the time of the invention to combine Cluet and Luotonen since both are concerned with providing a cache daemon with a directory server to improve performance as providing caching helps achieve fast performance and high availability. See page 273 or Cluet and column 1, lines 25-35 of Luotonen.

In reference to claim 7, Cluet teaches that the directory server is linked to a variety of network-based applications that store data and providing a cache storing those entries provides for efficient delivery of documents. Thus the directory server is not limited in its connections between various applications and a caching daemon. See page 273.

In reference to claims 9 and 10, Luotonen teaches that when a client requests a document, a proxy server determines if the file contained in the cache is up-to-date and delivers the document to the user if it is. If it is not up-to-date, the proxy server then it retrieves information from the origin server (directory server). See column 1, lines 39-50. Thus if the document were not stored in the cache originally, the proxy would then retrieve it from the remote server (directory server). The information is then stored in the proxy for future requests. See column 1. It would have been obvious to one of ordinary skill in the art at the time of the invention to combine Cluet and Luotonen since both are concerned with providing a cache daemon with a directory server to improve performance as providing caching helps achieve fast performance and high availability. See page 273 or Cluet and column 1, lines 25-35 of Luotonen.

Claims 16 and 17 are rejected under the same rationale used in claims 6 and 7 respectively above.

In reference to claims 18-20, Luotonen teaches that when a client requests a document, a proxy server determines if the file contained in the cache is up-to-date and delivers the document to the user if it is. If it is not up-to-date, the proxy server then it retrieves information from the origin server (directory server). See column 1, lines 39-

50. Thus if the document were not stored in the cache originally, the proxy would then retrieve it from the remote server (directory server). The information is then stored in the proxy for future requests. See column 1. It would have been obvious to one of ordinary skill in the art at the time of the invention to combine Cluet and Luotonen since both are concerned with providing a cache daemon with a directory server to improve performance as providing caching helps achieve fast performance and high availability. See page 273 or Cluet and column 1, lines 25-35 of Luotonen.

Claim 21 is rejected under the same rationale used in claim 1 above and further in view of the fact that Cluet teaches utilizing an LDAP caching daemon. See page 273, column 2 and page 276-277, "LDAP Directory Caches" of Cluet. It would have been obvious to a person of ordinary skill in the art at the time of the invention to utilize an LDAP caching daemon as taught by Cluet in conjunction with Luotonen's system since both are concerned with providing a cache daemon with a directory server in order to improve performance as providing a cache helps achieve fast performance and high availability. See page 273 or Cluet and column 1, lines 25-35 of Luotonen. Furthermore, providing a cache with a directory server not only improves performance but also avoids latency which is a major limitation when the cache is not connected to the directory server (as taught by both Cluet and Luotonen).

Claims 23-26 are rejected under the same rationale used above in claims 2-5 respectively.

Claim 27 is rejected under the same rationale used in claim 6 above and further in view of the fact that Cluet teaches utilizing an LDAP caching daemon. See page 273,

column 2 and page 276-277, "LDAP Directory Caches" of Cluet. It would have been obvious to a person of ordinary skill in the art at the time of the invention to utilize an LDAP caching daemon as taught by Cluet in conjunction with Luotonen's system since both are concerned with providing a cache daemon with a directory server in order to improve performance as providing a cache helps achieve fast performance and high availability. See page 273 or Cluet and column 1, lines 25-35 of Luotonen. Furthermore, providing a cache with a directory server not only improves performance but also avoids latency which is a major limitation when the cache is not connected to the directory server (as taught by both Cluet and Luotonen).

Claims 29-32 are rejected under the same rationale used above in claims 7-10 respectively.

Claim 33 is rejected under the same rationale used above in claim 16 and further in view of the fact that Cluet teaches utilizing an LDAP caching daemon. See page 273, column 2 and page 276-277, "LDAP Directory Caches" of Cluet. It would have been obvious to a person of ordinary skill in the art at the time of the invention to utilize an LDAP caching daemon as taught by Cluet in conjunction with Luotonen's system since both are concerned with providing a cache daemon with a directory server in order to improve performance as providing a cache helps achieve fast performance and high availability. See page 273 or Cluet and column 1, lines 25-35 of Luotonen. Furthermore, providing a cache with a directory server not only improves performance but also avoids latency which is a major limitation when the cache is not connected to the directory server (as taught by both Cluet and Luotonen).

Claims 35-38 are rejected under the same rationale used above in claims 17-20 respectively.

Claim 39 is rejected under the same rationale used in claim 1 above and further in view of the fact that Cluet teaches utilizing an LDAP caching daemon. See page 273, column 2 and page 276-277, "LDAP Directory Caches" of Cluet. It would have been obvious to a person of ordinary skill in the art at the time of the invention to utilize an LDAP caching daemon as taught by Cluet in conjunction with Luotonen's system since both are concerned with providing a cache daemon with a directory server in order to improve performance as providing a cache helps achieve fast performance and high availability. See page 273 or Cluet and column 1, lines 25-35 of Luotonen. Furthermore, providing a cache with a directory server not only improves performance but also avoids latency which is a major limitation when the cache is not connected to the directory server (as taught by both Cluet and Luotonen).

Claims 40 is rejected under the same rationale used in claims 3 and 4 above.

Claim 41 is rejected under the same rationale used in claim 5 above.

Claim 42 is rejected under the same rationale used in claim 16 above and further in view of the fact that Cluet teaches utilizing an LDAP caching daemon. See page 273, column 2 and page 276-277, "LDAP Directory Caches" of Cluet. It would have been obvious to a person of ordinary skill in the art at the time of the invention to utilize an LDAP caching daemon as taught by Cluet in conjunction with Luotonen's system since both are concerned with providing a cache daemon with a directory server in order to improve performance as providing a cache helps achieve fast performance and high

availability. See page 273 or Cluet and column 1, lines 25-35 of Luotonen. Furthermore, providing a cache with a directory server not only improves performance but also avoids latency which is a major limitation when the cache is not connected to the directory server (as taught by both Cluet and Luotonen).

Claims 43 is rejected under the same rationale used in claims 18 and 19 above.

Claim 44 is rejected under the same rationale used in claim 20.

(10) Response to Argument

Appellant generally argues with respect to all claims that the Cluet article is silent as to the number of connections and how long they are maintained. It is noted that the claim language recites "a plurality of connections" but does not recite limitations regarding how long the connections are maintained rather recites "simultaneously maintaining a plurality of connections". Cluet's system teaches that it is desirable to cache information in LDAP network directories to achieve fast performance, etc. Cluet teaches that a client cache stores a subset of the data available at the directory server. See abstract and introduction in which Cluet states, ***". . .it is desirable to cache information close to the applications that access the directory information. Such client caching has been established as an effective way to scale the performance of a client-server database architecture. Physical caching of directory entries occurs in LDAP directories."*** It is evident from Cluet's disclosure that he teaches maintaining a connection between the directory server and a caching daemon. Furthermore, Luotonen illustrates maintaining a connection between the directory server and the cache in figure 1. Luotonen further states that the invention includes cache

architecture that provides a preallocated structure of directories. See column 3, lines 55-67. Appellant further argues that Luotonen only teaches a connection to perform an "up-to-date" check when it is needed rather than maintain a connection. Examiner disagrees because Luotonen illustrates maintaining a connection between the directory server and the cache in figure 1. Luotonen further states that the invention includes cache architecture that provides a preallocated structure of directories. See column 3, lines 55-67. Examiner believes it is apparent from Luotonen's disclosure that the connection is continuously maintained in reference to the column and line numbers stated above and also in view of the "Background of the Invention" section. Luotonen teaches that it is "advantageous to provide a proxy server cache structure that stores and accesses documents in an optimum manner in a storage hierarchy that is easily managed". Furthermore, Luotonen identifies the shortcoming of prior art and achieves to cure those shortcomings in stating in column 2, "The ability to locate documents in a cache without latency induced by long path names and large directories is very important. It is also important to make it easy to clean up old cache documents . . ." In an effort to avoid latency, it is clear that Luotonen's connections are continuous.

With respect to claim 1, Appellant argues Cluet and Luotonen fail to teach, "simultaneously maintaining a first plurality of connections between the directory server and a caching daemon." Appellant further argues Cluet and Luotonen specifically teach away from reducing client server communications. Examiner respectfully disagrees. Cluet's system teaches that it is desirable to cache information in LDAP network directories to achieve fast performance, etc. Cluet teaches that a client cache stores a

subset of the data available at the directory server. See abstract and introduction in which Cluet states, ***". . .it is desirable to cache information close to the applications that access the directory information. Such client caching has been established as an effective way to scale the performance of a client-server database architecture. Physical caching of directory entries occurs in LDAP directories."*** It is evident from Cluet's disclosure that he teaches maintaining a connection between the directory server and a caching daemon. Furthermore, Luotonen illustrates maintaining a connection between the directory server and the cache in figure 1. Luotonen further states that the invention includes cache architecture that provides a preallocated structure of directories. See column 3, lines 55-67.

Appellant further argues that Cluet teaches away from simultaneously maintaining a plurality of connections because he teaches reducing client server communications which would be inconsistent with maintaining those connections. Examiner disagrees with Appellant's rationale because the purpose of utilizing a connection between the DIRECTORY server and cache is to reduce client server communications. Maintaining connections between a directory server and caching daemon is intended to minimize the time it takes to transfer documents from the ORIGINAL server.

Appellant arguments with respect to claims 21, 23-27, 29-33, and 35-44 are substantially similar to those above.

Appellant argues rejections with respect to claims 16-20, 33, 35-38 and 42-44 as being incorrectly rejected as non-statutory subject matter. A daemon is a process that

runs in the background and performs a specified operation at predefined times or in response to certain events and thus is directed to non-statutory subject matter since it does not define any structural and functional interrelationships between the computer program and other claimed elements of a computer which permit the computer program's functionality to be realized. A caching daemon is software per se.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

Rachna Singh

Conferees:

HEATHER R. HERNDON
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100

Heather Herndon
Supervisory Patent Examiner, AU 2176

William Bashore
Primary Patent Examiner, AU 2176

WILLIAM BASHORE
PRIMARY EXAMINER
2/5/2006